

**DETAILED ACTION**

1. This communication is responsive to the amendment filed 07/08/2008 and the telephonic interview on 09/24/2008.

Claims 1, 3, 6-8, 10, 13-15, 17, 21, and 21 have been examined and allowed.

2. **EXAMINER'S AMENDMENT:**

An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Mr. Stanley J. Gradisar (Registration No. 42, 598) on 09/24/2008.

**The application has been amended as follows:**

**In the Claims:**

**This listing of claims will replace all prior versions, and listings, of claims in the application:**

1. (Currently Amended) A computer-implemented method of resource lookup comprising:

generating a code by compiling an application source file and a project file of the application source file;

receiving a relative resource identifier from the application source file indicating a resource to be utilized by the application, wherein the relative resource identifier does not indicate a protocol or a location for the resource;

locating the resource based on the relative resource identifier and the code generated during compilation of the application; and

returning the resource to the application,

wherein the code generated during compilation of the application comprises a branching construct having one or more cases,

wherein each case of the branching construct comprises resource information identifying the resource indicated by the relative resource identifier, and

wherein receiving the relative resource identifier from the application source file comprises receiving the relative resource identifier via an Application Program Interface.

2. (Cancelled)

3. (Currently Amended) The method of claim [[2]] 1, wherein the relative resource identifier is a string representing a name of the resource.

4. -5. (Cancelled)

6. (Original) The method of claim 1, wherein returning the resource to the application comprises returning an object that is an instance of a class of the resource.

7. (Original) The method of claim 1, wherein returning the resource comprises returning an open stream to the resource.

8. (Currently Amended) A system for resource lookup comprising:

a processor; and

~~a memory coupled with and readable by the processor and containing a series of instructions that, when executed by the processor, cause the processor to generate a code by compiling an application source file and a project file of the application source file and to receive a relative resource identifier from the application source file indicating a resource to be utilized by the application, wherein the relative resource identifier does not indicate a protocol or a location for the resource, and to locate the resource based on the relative resource identifier and the code generated during compilation of the application, and return the resource to the application;~~

a memory coupled with and readable by the processor and containing a series of instructions that, when executed by the processor, cause the processor to:

generate a code by compiling an application source file and a project file

of the application source file;

receive a relative resource identifier from the application source file  
indicating a resource to be utilized by the application, wherein the relative resource  
identifier does not indicate a protocol or a location for the resource;

locate the resource based on the relative resource identifier and the code  
generated during compilation of the application, and

return the resource to the application,

wherein the code generated during compilation of the application comprises a  
branching construct having one or more cases,

wherein each case of the branching construct comprises resource information  
identifying the resource indicated by the relative resource identifier, and

wherein receiving the relative resource identifier from the application source file  
comprises receiving the relative resource identifier via an Application Program Interface.

9. (Cancelled)

10. (Currently Amended) The system of claim [[9]] 8, wherein the relative resource identifier is a string representing a name of the resource.

11. -12.(Cancelled)

13. (Original) The system of claim 8, wherein returning the resource to the application comprises returning an object that is an instance of a class of the resource.

14. (Original) The system of claim 8, wherein returning the resource comprises returning an open stream to the resource.

15. (Currently Amended) A machine-readable storage medium encoding a computer program of instructions for executing a computer process for resource lookup by a computer system, said computer process comprising:

generating a code by compiling an application source file and a project file of the application source file;

receiving a relative resource identifier from the application source file indicating a resource to be utilized by the application, wherein the relative resource identifier does not indicate a protocol or a location for the resource;

locating the resource based on the relative resource identifier and the code generated during compilation of the application; and

returning the resource to the application,

wherein the code generated during compilation of the application comprises a branching construct having one or more cases,

wherein each case of the branching construct comprises resource information identifying the resource indicated by the relative resource identifier, and

wherein receiving the relative resource identifier from the application source file  
comprises receiving the relative resource identifier via an Application Program Interface.

16. (Cancelled)

17. (Currently Amended) The machine-readable storage medium of claim [[16]] 15,  
wherein the relative resource identifier is a string representing a name of the resource.

18.-19. (Cancelled)

20. (Previously Presented) The machine-readable storage medium of claim 15, wherein  
returning the resource to the application comprises returning an object that is an instance  
of a class of the resource.

21. (Previously Presented) The machine-readable storage medium of claim 15, wherein  
returning the resource comprises returning an open stream to the resource.

3. **REASONS FOR ALLOWANCE:**

Claims 1, 3, 6-8, 10, 13-15, 17, 21, and 21 are allowed.

The following is an examiner's statement of reasons for allowance:

Interpreting the claims in light of the specification, Examiner finds the claimed invention is patentably distinct from the prior art of record.

The prior art does not expressly teach or render obvious the invention as recited in independent claims 1, 8, and 15.

The features as recited in independent claims 1, 8, and 15 “*locating the resource based on the relative resource identifier and the code generated during compilation of the application; wherein the code generated during compilation of the application comprises a branching construct having one or more cases, wherein each case of the branching construct comprises resource information identifying the resource indicated by the relative resource identifier, and wherein receiving the relative resource identifier from the application source file comprises receiving the relative resource identifier via an Application Program Interface*

”, when taken in the context of the claims as a whole, were not uncovered in the prior art teachings.

Dependent claims are allowed as they depend upon allowable independent claims.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled “Comments on Statement of Reasons for Allowance.”

## **CONTACT INFORMATION**

Any inquiry concerning this communication or earlier communications from the examiner should be directed to VAN H. NGUYEN whose telephone number is (571) 272-3765. The examiner can normally be reached on Monday-Thursday from 8:30AM - 6:00PM. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, MENG-AI AN can be reached at (571) 272-3756.

The fax phone number for the organization where this application or proceeding is assigned is **571-273-8300**.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

**/VAN H NGUYEN/  
Primary Examiner, Art Unit 2194**